ccache Compiler cache

CONCEPTS

ccache hashes the preprocessed source, compiler and flags. On a hit it returns the stored object file instead of compiling. Run a build twice and compare 'ccache -s' to see it working.

ccache gcc ... hash inputs
↓ lookup
Cache hit: copy object
↓ miss
Real compiler store result

Each compile is a lookup keyed by source, flags and compiler

RUNNING IT

ccache *<compiler> <args>*
Prefix a compiler command

ln -s $(which ccache) ~/bin/gcc
Masquerade as the compiler

make CC='ccache gcc'
Pass the prefix to make

CC="ccache gcc" ./configure
Use it with autotools

CMAKE_C_COMPILER_LAUNCHER
Set to ccache via -D (also CXX)

ccache *<KEY>*=*<value>* **gcc ...**
Set one config option for this call

STATISTICS

ccache -s
Show summary of hits and misses

ccache -s -v
Show more detailed statistics

ccache -z
Zero the statistics counters

CACHE MANAGEMENT

ccache -M *<size>*
Set maximum cache size (default GiB)

ccache -F *<num>*
Set maximum number of files

CACHE MANAGEMENT (cont.)

ccache -c
Clean up to the configured limits

ccache -C
Clear all cached files, keep config

ccache --evict-older-than *<age>***d**
Drop entries unused for N days

ccache -d *<dir>* **-s**
Operate on another cache directory

CONFIGURATION

ccache -p
Show effective config and its origin

ccache -k *<key>*
Print one config value

ccache -o *<key>*=*<value>*
Write a config option to the file

CCACHE_DIR=*<dir>*
Cache location

CCACHE_MAXSIZE=*<size>*
Cache size limit (default 5GiB)

CCACHE_DISABLE=1
Bypass ccache entirely

HIT RATE AND CORRECTNESS

CCACHE_BASEDIR=*<dir>*
Rewrite absolute paths under dir

CCACHE_SLOPPINESS=*<list>*
Relax checks to get more hits

CCACHE_SLOPPINESS=time_macros
Ignore __DATE__ and __TIME__

CCACHE_SLOPPINESS=pch_defines
Allow precompiled headers

CCACHE_COMPILERCHECK=content
Hash compiler binary contents

DEBUGGING AND SHARING

CCACHE_LOGFILE=*<file>*
Log every decision to a file